

**REMARKS**

The present Amendment amends claims 3, 6 , 8, 9 and 17, cancels claims 1, 2, 4, 5, 13-16 and 21 and leaves claims 10-12 and 18-20 unchanged. Therefore, the present application has pending claims 3, 6, 8-12 and 17-20.

**Interview Summary**

Applicants thank the Examiner for granting the interview conducted on November 16, 2007. In the interview, arguments were presented to overcome the cited reference, particularly Bala. The Examiner and Applicants' representative did not come to an agreement with regard to any of the arguments presented.

In this response, Applicants have reiterated the arguments made during the interview, and respectfully request the Examiner's full reconsideration of such arguments, as provided in more detail in this response.

**35 U.S.C. §102 Rejections**

Claims 1-21 stand rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent Application Publication No. 2002/0184618 to Bala, et al. ("Bala"). As indicated previously, claims 1, 2, 4, 5, 13-16 and 21 were canceled. Therefore, this rejection regarding claims 1, 2, 4, 5, 13-16 and 21 is rendered moot. Regarding the remaining claims 3, 6, 8-12 and 17-20, this rejection is traversed for the following reasons. Applicants submit that the features of the present invention as now more clearly recited in claims 3, 6, 8-12 and 17-20 are not taught or suggested by Bala, whether taken individually or in combination any of the other references of record. Therefore, Applicants respectfully request the Examiner to reconsider and withdraw this rejection.

Amendments were made to the claims to more clearly describe features of the present invention. Specifically, amendments were made to the claims to more clearly recite that the present invention is directed to a data format conversion method, data format conversion equipment, a controller, and a controller management system as recited, for example, in independent claims 3, 6, 9, 17, and 18.

Claims 3 and 6-8

The present invention, as recited in claim 3, and as similarly recited in claim 6, provides a data format conversion method. The method includes inputting information including a convert direction for converting a program of a first data format to a program of a second data format. The method also includes converting the program of the first data format to the program of the second data format, based on a conversion rule designated by the convert direction of a data format contained in the information. According to the present invention, the information further includes the program of the first data format, and the conversion rule is acquired through a communication network. The prior art does not disclose all of these features.

The above described features of the present invention, as now more clearly recited in the claims, are not taught or suggested by any of the references of record, particularly Bala, whether taken individually or in combination with any of the other references of record.

Bala teaches networked client-server architecture for transparently transforming and executing application. However, there is no teaching or suggestion

in Bala of the data format conversion method or the data format conversion equipment as recited in claims 3 and 6 of the present invention.

Bala provides for native execution of an application on a client using code segments transmitted from a server over a network. The server includes an application code source, and a server code segment manager. The server may also include an application code transformation manager if the code source is not in the native binary format of the client. The client includes a client code segment manager, a code cache linker and manager, a code cache, and a CPU. When the client seeks to execute an application, code segments are transmitted from the server to the client and are stored in the code cache. The CPU then executes the code segments natively. When a code segment branches to a segment not in the cache, control passes to the client code segment manager, which requests the needed code segment from the server code segment manager of the server.

One feature of the present invention, as recited in claim 3 includes converting the program of the first data format to the program of the second data format, based on a conversion rule designated by the convert direction of a data format contained in the information, where the information further includes the program of the first data format, and the conversion rule is acquired through a communication network. Bala does not disclose this feature. For example, Bala does not teach or suggest where the conversion rule is acquired through a communication network. To support the assertion that Bala teaches this feature, the Examiner cites paragraphs [0035] to [0037]. However, the cited text describes the client-server configuration illustrated in Fig. 1, which includes an application code transformation manager 20 that is responsible for transforming segments of application code source 16 into binary

code segments that can be executed natively by the client. There is no teaching or suggestion in the cited text, or any other portions of Bala, of acquiring a conversion rule in the manner claimed.

In the present invention, a “conversion rule” (program for converting from a “first data format” into a “second data format”) is obtained from another apparatus through a “communication network”. Accordingly, even if “conversion rule” does not exist in the “data format conversion apparatus”, data conversion can be achieved by way of obtaining the conversion rule from another apparatus through the “communication network.” Furthermore, even if a user of the controller does not keep retaining conversion program (rule) locally, a conversion program from a web server of the maker can be used. This is quite different from Bala.

Therefore, Bala fails to teach or suggest converting the program of the first data format to the program of the second data format, based on a conversion rule designated by the convert direction of a data format contained in the information, wherein the information further includes the program of the first data format, and wherein said conversion rule is acquired through a communication network” as recited in claim 3, and as similarly recited in claim 6.

### Claims 9-12

The present invention, as recited in claim 9, provides Data format conversion equipment comprising a computer. The computer includes communication means for acquiring input information containing a program, etc., of a first data format from a communication network. The computer also includes direction extract means for extracting a convert direction of a data format contained in the input information so

acquired. Also included in the computer is data conversion rule acquisition means for acquiring a conversion program for executing conversion to a data format corresponding to the convert direction so extracted from outside through the communication means. The computer further includes convert execution means for converting the program of the first data format to a program of a second data format on the basis of the conversion program so acquired, and outputting the converted program to the communication network through the communication means. According to the present invention, the conversion program converts the program to a program of a third data format different from the first and second data formats, and then converts the program to the second data format. The prior art does not disclose all of these features.

The above described features of the present invention, as now more clearly recited in the claims, are not taught or suggested by any of the references of record, particularly Bala, whether taken individually or in combination with any of the other references of record.

As previously discussed, Bala teaches networked client-server architecture for transparently transforming and executing application. However, there is no teaching or suggestion in Bala of the data format conversion equipment as recited in claim 9 of the present invention.

One feature of the present invention, as recited in claim 9 includes where the conversion program converts the program to a program of a third data format different form the first and second data formats, and then converts the program to the second data format. Bala does not disclose this feature. For example, Bala merely discloses that a server generates native binary code in accordance with a

CPU type of client. There is no teaching or suggestion in Bala of a converting the program to a program of a third data format, in the manner claimed.

Furthermore, a distinguishing feature of claim 9 includes generating a third data format, such as a universal format (e.g., XML). Bala does not disclose generating the third data format as in this universal format, at all. To the contrary, Bala only discloses a SERVER generating native binary code in accordance with CPU TYPE of CLIENT. This is quite different from the present invention.

Another feature of the present invention, as recited in claim 11, includes where the convert execution means includes a virtual machine and intermediate conversion means for converting a program to a third data format. Bala does not disclose this feature. Bala only discloses an Application Code Transformation Manager, which is executed by a server (as shown in Fig. 2B, for example). This is not the same as a virtual machine, as claimed.

Yet another feature of the present invention, as recited in claim 12, includes where the virtual machine is a virtual machine of Java. Bala does not disclose where a virtual machine, as in the present invention, is a virtual machine of Java.

Therefore, Bala fails to teach or suggest "wherein said conversion program converts the program to a program of a third data format different from the first and second data formats, and then converts the program to the second data format" as recited in claim 9.

Furthermore, Bala fails to teach or suggest "wherein said convert execution means comprises a virtual machine and intermediate conversion means for converting a program etc, to a third data format" as recited in claim 11.

Even further, Bala fails to teach or suggest "wherein said virtual machine is a

virtual machine of Java" as recited in claim 12.

Claims 17-20

The present invention, as recited in claim 17, and as similarly recited in claim 18, provides a controller, including a computer. The controller controls a controlled target in accordance with a predetermined program of a second data format. The computer includes memory means for storing a program of a first data format inputted through a communication network, and a function of outputting the program of the first data format stored in the memory means in response to a request input through the communication network. The prior art does not disclose all of these features.

The above described features of the present invention, as now more clearly recited in the claims, are not taught or suggested by any of the references of record, particularly Bala, whether taken individually or in combination with any of the other references of record.

As previously discussed, Bala teaches networked client-server architecture for transparently transforming and executing application. However, there is no teaching or suggestion in Bala of the controller or the controller management system as recited in claims 17 and 18 of the present invention.

One feature of the present invention, as recited in claim 17 includes where the computer of the controller includes memory means for storing a program of a first data format input through a communication network, and a function of outputting the program of the first data format stored in the memory means in response to a request input through the communication network. That is to say, the controller of

the present invention includes program data for controlling itself. Accordingly, it is not necessary to contain original programs in other storage media. More specifically, in the present invention, the controller itself receives a program of a “first data format” and sends it on request. That is, the controller stores program data which can be comprehended by a program generating means, for controlling itself. With use of this structure, since the program to be executed by the controller is stored in a format which can be comprehended by the “programming generation means”, it is not necessary to keep storing the original program at another storing medium. Therefore, maintainability will be improved. Bala does not disclose this feature of the present invention. In Bala, for overcoming problems relating to security, Application Code Source 18 is stored in the SERVER as disclosed in [0032], but there is no disclosure in Bala of storing the source in the CLIENT.

Another feature of the present invention, as recited in claim 18, includes a data format converter for inputting the program of the first data format created by the programmer unit, for converting the program to a program of a second data format, and for outputting the program to said controller. As previously discussed, Bala does not disclose this feature. More specifically, Bala does not teach or suggest where the controller of the present invention includes program data for controlling itself, in the manner claimed.

Also, in claim 18 of this invention, the controller itself stores the above-mentioned “programming generation means (programmer unit)”. Therefore, it is unnecessary to store the “programming generation means” in another storing medium. Bala does not disclose this feature. To the contrary, Bala merely discloses where Application Code Source 18 is stored in the server, but it does not disclose

where the source is provided to the client (see, for example, paragraphs [0032] and [0010] of Bala).

Therefore, Bala fails to teach or suggest "memory means for storing a program of a first data format input through a communication network; and a function of outputting the program of the first data format stored in said memory means in response to a request input through said communication network" as recited in claim 17.

Furthermore, Bala fails to teach or suggest "a data format converter for inputting the program of the first data format created by said programmer unit, converting the program to a program of a second data format and outputting the program to said controller" as recited in claim 18.

Therefore, Bala does not teach or suggest the features of the present invention, as recited in claims 3, 6, 8-12 and 17-20. Accordingly, reconsideration and withdrawal of the 35 U.S.C. §102(e) rejection of claims 3, 6, 8-12 and 17-20 as being anticipated by Bala are respectfully requested.

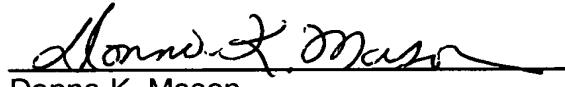
The remaining references of record have been studied. Applicants submit that they do not supply any of the deficiencies noted above with respect to the references used in the rejection of claims 3, 6, 8-12 and 17-20.

In view of the foregoing amendments and remarks, Applicants submit that claims 3, 6, 8-12 and 17-20 are in condition for allowance. Accordingly, early allowance of claims 3, 6, 8-12 and 17-20 is respectfully requested.

To the extent necessary, Applicants petition for an extension of time under 37 CFR 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including extension of time fees, or credit any overpayment of fees, to the deposit account of Mattingly, Stanger, Malur & Brundidge, P.C., Deposit Account No. 50-1417 (referencing attorney docket no. 500.42830X00).

Respectfully submitted,

MATTINGLY, STANGER, MALUR & BRUNDIDGE, P.C.



Donna K. Mason  
Donna K. Mason  
Registration No. 45,962

DKM/cmd  
(703) 684-1120